AMENDMENTS TO THE CLAIMS

1. (Canceled)

2. (Canceled)

3. (Currently amended) A method utilizing a graphical user interface in a computer system, comprising the steps of:

    executing an application program with a graphical user interface comprising a plurality of elements, each ~~said~~ element being associated with a set of commands; and

    changing the graphical representation of one or more of the ~~said~~ elements when two or more of <u>the</u> ~~said~~ elements are <u>disposed within</u> [[in]] close proximity <u>of each other</u>.

4. (New) The method of claim 3, wherein each element comprises a core and a dynamic edge surrounding the core.

5. (New) The method of claim 4, wherein changing the graphical representation when two or more elements are disposed within close proximity comprises joining the elements together to form a group of elements if the edges of the elements overlap.

6. (New) The method of claim 5, further comprising:

    receiving a user input to move an element in a group of joined elements; and

    separating the moved element from the group when the separated element is moved out of proximity from the group of joined elements.

7.   (New) The method of claim 5, further comprising:

receiving a user input to move a group of joined elements; and

repositioning the group of joined elements within the user interface according to the

user input, preserving the spatial relationship among the joined elements.

8.   (New) The method of claim 3, wherein each element has a color, and changing
the graphical representation of two or more elements of different color comprises forming an
overlapping region between the two or more elements, the color of the overlapping region
derived from the colors of each of the two or more elements.

9.   (New) A computer-implemented method for adjusting a graphical user interface
of a computer program, the method comprising:

displaying a plurality of elements, each element associated with at least one function

of the computer program;

receiving user inputs to move elements within the user interface; and

in response to receiving a user input, moving a first element to a position overlapping

a second element and merging the first and second elements to form a group.

10.   (New) The method of claim 9, further comprising:

receiving a user input to move the group of the first and second elements within the

user interface; and

in response to a reception of the user input to move the group, repositioning the group

of elements within the user interface while preserving the spatial relationship

among the merged elements of the group.

11.   (New) The method of claim 9, further comprising:

in response to a reception of a user input, moving a third element to a position

overlapping the group of first and second elements and merging the third

element with the first and second elements to form a new group.

12.   (New) The method of claim 11, further comprising:

   responding to a user input by moving the third element to a position not overlapping the first and second elements and removing the third element from the group of merged elements to reform the group.

13.   (New) The method of claim 9, wherein each element comprises:

   a functional core region that is selectable by a user to invoke the one or more functions associated with the element; and

   a dynamic edge region operative to change shape when the element is merged with one or more other elements, the dynamic edge region forming a continuous dynamic edge region with the other elements merged therewith.

14.   (New) The method of claim 9, wherein each element has at least one color, and each group of merged elements includes an overlapping region between the elements, the color of the overlapping region derived from the colors of each of the elements that form the overlapping region.

15.   (New) The method of claim 9, wherein the first and second elements are of a different color, the method further comprising:

   a step for fusing the colors of the first and second elements in an overlapping region therebetween.

16.   (New) A computer program product for adjusting a graphical user interface of a computer program, the computer program product comprising a computer-readable medium containing computer program code for performing the operations:

   displaying a plurality of elements, each element associated with at least one function of the computer program;

   receiving user inputs to move elements within the user interface; and

   responding to a reception of a user input indicating a move of a first element to a position overlapping a second element by merging the first and second elements to form a group of elements.

17.   (New) The computer program product of claim 16, further comprising:

receiving a user input to move the group of the first and second elements within the user interface; and

in response to a reception of a user input to move the group, repositioning the group of elements within the user interface while preserving the spatial relationship among the merged elements of the group.

18.   (New) The computer program product of claim 16, the computer program code further for performing the operations:

in response to a reception of a user input to move a third element to a position overlapping the group of first and second elements, merging the third element with the first and second elements to reform the group.

19.   (New) The computer program product of claim 18, the computer program code further for performing the operations:

in response to a reception of a user input to move the third element to a position not overlapping the first and second elements, removing the third element from the group of merged elements to reform the group.

20.   (New) The computer program product of claim 16, wherein each element comprises:

a functional core region that is selectable by a user to invoke the one or more functions associated with the element; and

a dynamic edge region that changes shape when the element is merged with one or more other elements, the dynamic edge region forming a continuous dynamic edge region with the other elements merged therewith.

21.   (New) The computer program product of claim 16, wherein each element has at least one color, and each group of merged elements includes an overlapping region between the elements, the color of the overlapping region derived from the colors of each of the elements that form the overlapping region.

22.   (New) The computer program product of claim 16, wherein the computer program product is a media rendering software application.